

RECEIVED
CENTRAL FAX CENTER
MAR 23 2007LAW OFFICES OF
FAY KAPLUN & MARCIN, LLP
INTELLECTUAL PROPERTY LAW150 BROADWAY, SUITE 702
NEW YORK, NEW YORK 10038
PHONE: (212) 619-6000
FAX : (212) 208-6819
WWW.FKMIPLAW.COM**FACSIMILE COVER SHEET**

FAX NO : (571) 273-8300

TO : Mail Stop: Appeal-Brief Patent
Commissioner for Patents

FROM : Michael J. Marcin, Esq. of Fay Kaplun & Marcin, LLP

DATE : March 23, 2007

SUBJECT : U.S. Patent Appln. Serial No. 09/920,995
for *System and Method for Implementing a Smart System Call*
Your Ref: 2000.023
Our Reference: 40101/08201

NUMBER OF PAGES INCLUDING COVER : 18

MESSAGE:

Please see attached.

Thank you.

IF ANY PAGES WERE NOT RECEIVED OR ARE ILLEGIBLE, PLEASE CALL (212) 619-6000 AS SOON AS POSSIBLE


The information contained in this facsimile message is attorney privileged and confidential information intended only for the use of the individual or entity named above. If the reader of this message is not the intended recipient or the employee or agent responsible to deliver it to the intended recipient, you are hereby notified that any dissemination, distribution or copying of this communication is strictly prohibited. If you have received this communication in error, please notify us by telephone, and return the original message to us at the above address via the U.S. Postal Service. We will reimburse any costs you incur in notifying us and returning the message to us. Thank you.

Attorney Docket No. 40101/08201 (2000.023)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**RECEIVED
CENTRAL FAX CENTER****MAR 23 2007**

Inventor(s) : Nunoe
Serial No. : 09/920,995
Filing Date : August 1, 2001
For : System and Method for Implementing a Smart System Call
Group Art Unit : 2194
Confirmation No. : 5812
Examiner : Charles E. Anya

Mail Stop: Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Certificate of Facsimile/Mailing	
I hereby certify that this correspondence is being deposited via facsimile of First Class Mail addressed to:	
Mail Stop: Appeal Brief-Patents Commissioner for Patents P.O. Box 1450 Alexandria, VA 22313-1450 571-273-8300	
By: 	Date: March 23, 2007
Michael J. Marcin, Reg. No. 48,198	

TRANSMITTAL

In response to the Final Office Action dated September 6, 2006 and the Advisory Action dated January 23, 2007, transmitted herewith please find a Notice of Appeal for filing in the above-identified application. Please charge the Credit Card of Fay Kaplun & Marcin, LLP in the amount of \$500.00 (PTO-Form 2038 is enclosed herewith). The Commissioner is hereby authorized to charge the **Deposit Account of Fay Kaplun & Marcin, LLP NO. 50-1492** for any additional required fees. A copy of this paper is enclosed for that purpose.

Respectfully submitted,

Dated: March 23, 2007

By: 
Michael J. Marcin, Reg. 48,198

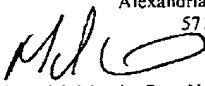
Fay Kaplun & Marcin, LLP
150 Broadway, Suite 702
New York, NY 10038
Tel: (212) 619-6000
Fax: (212) 619-0276

Attorney Docket No. 40101/08201 (2000.023)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**RECEIVED
CENTRAL FAX CENTER****MAR 23 2007**

Inventor(s) : Nunoe
Serial No. : 09/920,995
Filing Date : August 1, 2001
For : System and Method for Implementing a Smart System Call
Group Art Unit : 2194
Confirmation No. : 5812
Examiner : Charles E. Anya

Mail Stop: Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Certificate of Facsimile/Mailing	
I hereby certify that this correspondence is being deposited via facsimile of First Class Mail addressed to:	
Mail Stop: Appeal Brief-Patents Commissioner for Patents P.O. Box 1450 Alexandria, VA 22313-1450 571-273-8300	
By: 	Date: March 23, 2007
Michael J. Marcin, Reg. No. 48,198	

TRANSMITTAL

In response to the Final Office Action dated September 6, 2006 and the Advisory Action dated January 23, 2007, transmitted herewith please find a Notice of Appeal for filing in the above-identified application. Please charge the Credit Card of Fay Kaplun & Marcin, LLP in the amount of \$500.00 (PTO-Form 2038 is enclosed herewith). The Commissioner is hereby authorized to charge the **Deposit Account of Fay Kaplun & Marcin, LLP NO. 50-1492** for any additional required fees. A copy of this paper is enclosed for that purpose.

Respectfully submitted,

Dated: March 23, 2007

By: 
Michael J. Marcin, Reg. 48,198

Fay Kaplun & Marcin, LLP
150 Broadway, Suite 702
New York, NY 10038
Tel: (212) 619-6000
Fax: (212) 619-0276

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

**RECEIVED
CENTRAL FAX CENTER**

MAR 23 2007

In re Application of:

Hdei Nunoe

Serial No.: 09/920,995

Filed: August 1, 2001

For: SYSTEM AND METHOD FOR
IMPLEMENTING A SMART
SYSTEM CALL

Group Art Unit: 2194

Examiner: Charles E. Anya

**Board of Patent Appeals and
Interferences**

Mail Stop: Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF UNDER 37 C.F.R. § 41.37

In support of the Notice of Appeal filed January 23, 2007, and pursuant to 37

C.F.R. § 41.37, Appellant presents their appeal brief in the above-captioned application.

This is an appeal to the Board of Patent Appeals and Interferences from the Examiner's final rejection of claims 1-7 in the Final Office Action dated September 6, 2006. The appealed claims are set forth in the attached Claims Appendix.

03/26/2007 TL0111 00000017 09920995

01 FC:1402

500.00 0P

RECEIVED
CENTRAL FAX CENTERSerial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

MAR 23 2007

1. Real Party in Interest

This application is assigned to Wind River Systems, Inc., the real party in interest.

2. Related Appeals and Interferences

There are no other appeals or interferences which would directly affect, be directly affected, or have a bearing on the instant appeal.

3. Status of the Claims

Claims 1-7 have been rejected in the final Office Action. The final rejection of claims 1-7 is being appealed.

4. Status of Amendments

All amendments submitted by the appellant have been entered. None were submitted after the Advisory Action.

5. Summary of Claimed Subject Matter

The present invention relates to a system and method for implementing a "smart system call" that is capable of making function calls to privileged memory locations. (See Specification, p. 2, ¶ [0013]). Specifically, claim 1 of the present invention describes a method for determining a current processing mode (102, 302, 402) an executing software function. (See Id., p. 2, ¶ [0015]; p. 3, ¶¶ [0024], [0026]; and Figs. 1, 3, and 4). The method further comprises providing either direct or indirect access to a software function by determining the current processing mode of). (See Id.). The processing mode is defined as either an unprivileged

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

processing mode (108, 308, 408) of protection having a lower level access right to the memory of a computer system, or a privileged processing mode (106, 306, 406) of protection having a higher level access right to the memory. (See Id.).

In another embodiment, claim 6 of the present invention relates to a method comprising identifying a program code segment implementing an access to a memory area to be executed within a privileged processing mode (202). (See Id., p. 2, ¶ [0019]; and Fig. 2). The method further comprises replacing the program code segment with a substitute code segment. (See Id., p. 2, ¶ [0019]; and p. 3, ¶ [0028]). The substitute code segment includes program code to identify a current processing mode of the program code segment, execute a direct program flow control instruction if the current processing mode is the privileged processing mode (204), and execute an indirect program flow control instruction if the current processing mode is an unprivileged processing mode (206). (See Id.).

A further embodiment, claim 7 of the present invention relates to a computer readable medium encoded with a software application, comprising a software code implementing application functionality; and a smart system call into an operating system; wherein the smart system call comprises the software code to identify a current processing mode (102, 302, 402) of a program code segment. (See Id., p. 2, ¶ [0015]; p. 3, ¶¶ [0024], [0026]; and Figs. 1, 3, and 4). The smart system call further comprises software code to execute a direct program flow control instruction if the current processing mode is a privileged processing mode (106, 306, 406), and execute an indirect program flow control instruction if the current processing mode is an unprivileged processing mode (108, 308, 408). (See Id.).

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

6. Grounds of Rejection to be Reviewed on Appeal

I. Whether claims 1-7 are unpatentable under 35 U.S.C. § 103(a) as obvious over U.S. Patent No. 6,175,916 to Ginsberg et al. ("the Ginsberg patent") in view of U.S. Patent No. 6,542,919 to Wendorf et al. ("the Wendorf patent").

7. Argument

- I. The Rejection of Claims 1-7 Under 35 U.S.C. § 103(a) as Being Obvious Over U.S. Patent No. 6,175,916 to Ginsberg et al. in View of U.S. Patent No. 6,542,919 to Wendorf et al. Should Be Reversed.

A. The Examiner's Rejection

In the 09/06/06 Final Office Action, the Examiner rejected claims 1-7 Under 35 U.S.C. § 103(a) as being unpatentable over the Ginsberg patent in view of the Wendorf patent. (See 09/06/06 Office Action, p. 2, ¶ 3).

The Ginsberg patent relates to a method of making a call from one process to another, wherein the method includes executing a jump instruction from a first process, while specifying an invalid destination such as an odd virtual memory address. (See the Ginsberg patent, Abstract). Specifically, the Ginsberg patent describes a system that includes a processor and a memory fault handler. (See *Id.*, col. 7, lines 4-11). The memory fault handler is described as a privileged-mode component that can change virtual memory mapping when a jump function specifies an invalid address as the destination address. (See *Id.*). In order to change the virtual memory mappings, the processor must switch to privileged mode. (See *Id.*, col. 6, lines 63-65). If the invalid address is an odd address, then the memory fault handler recognizes that a call to a system function was attempted and executes a call processing function using the invalid address

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

as an index into tables of available system functions. (See Id., col. 7, lines 12-20). Therefore, according to the disclosure of the Ginsberg patent, the memory fault handler, a privileged-mode component, executes a call processing function if the invalid address is an odd address. Thus, the memory fault handler executes a function that associates the invalid address with the table of available system functions.

The Wendorf patent relates to an operating system wherein the provided protection domain support is arranged so as to be compatible with threads that obtain all their memory allocations from the operating system and were written without regard for protection domains. (See the Wendorf patent, Abstract). Specifically, the Wendorf patent relates to a process for setting a value of a protection domain register when there is a context switch (i.e., a switch from one thread to another). (See Id., col. 6, line 62 – col. 7, line 17). Upon the occurrence of a context switch, a test is performed to determine if the thread whose execution is being initiated is in the protection of the operating system. (See Id.).

B. The Cited Patents Do Not Disclose When the Current Processing Mode is an Unprivileged Processing Mode, Executing an Indirect Program Flow Control Instruction to Cause Execution of the Instruction within Software Having the Privileged Processing Mode and Determining a Current Processing Mode of an Executing Software, and as Recited in Claim 1.

According to the exemplary embodiments of the present invention, the “smart system call” includes a code stub that may be used to determine the current mode of execution and determine where the current mode is privileged (i.e., allowed direct access to the desired memory address). (See Specification, p. 2, ¶ [0013]). Specifically, claim 1 recites a “method comprising determining a current processing mode of an executing software function...” If the current mode is unprivileged, an alternate indirect system call mechanism may be used to allow

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

for indirect access to the desired memory address. (See Id.). Claim 1 further recites, "...when the current processing mode is an unprivileged processing mode, executing an indirect program flow control instruction to cause execution of the instruction within the software having the privileged processing mode." This mechanism may be an interrupt or exception based system call that may cause the spawning of privileged mode tasks in order to execute the desired function. (See Id.). Claim 1 further recites, "...when the current processing mode is a privileged processing mode, executing a direct program flow control instruction to directly access an instruction within a software having the privileged processing mode..."

According to an exemplary code stub of the present invention, if a current code is not unprivileged, then a direct jump (or change in program control) is made to the entry address of the desired function. (See Id., p. 2, ¶ [0015]). Conversely, if the current mode is unprivileged, then the system call is made using the interrupt based system call. (See Id.). Therefore, the present invention may allow for easy implementation of direct access to protected memory in an operating system that uses indirect system calls, such as interrupt-based system calls or exception-based system calls. (See Id.). In other words, this exemplary code stub may be used to emulate indirect system calls in an operating system that does not support such calls. As noted above, claim 1 recites, that "when the current processing mode is an unprivileged processing mode, executing an indirect program flow control instruction to cause execution of the instruction within the software having the privileged processing mode."

In contrast to the present invention, the Ginsberg patent fails to teach or suggest the execution of an indirect program flow control instruction to cause execution of the instruction within software having the privileged processing mode *when the current processing mode is an unprivileged processing mode*. The Examiner asserts that the performance of the memory fault

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

handler of the Ginsberg patent to “when the current processing mode is an unprivileged processing mode, executing an indirect program flow control instruction,” as recited in claim 1 of the present invention. (See 09/06/06 Office Action, p. 2, ¶ 4). However, the Ginsberg patent explicitly states that the memory fault handler is a privileged-mode component, and in order for the memory fault handler to perform (i.e., change virtual memory mappings), the processor must switch to privileged mode. (See the Ginsberg patent, col. 6 lines 63-65; col. 7, lines 4-11). According to the Ginsberg patent, the memory fault handler responds to invalid addresses, or memory faults, and performs system critical operations such as changing virtual memory mappings. (See *Id.*, col. 7 lines 29-36). However, since the memory fault handler is a privileged-mode component, any functions executed by the memory fault handler will only be performed when the current processing mode is a privileged processing mode. Therefore, it is respectfully submitted that the Ginsburg disclosure teaches away from the recitations in claim 1, which state, *inter alia*, “...when the current processing mode is an unprivileged processing mode, executing an indirect program flow control instruction to cause execution of the instruction within the software having the privileged processing mode.” (Emphasis added). Accordingly it would not be possible for the memory fault handler of the Ginsberg patent to execute an indirect program flow control instruction, when the current processing mode is an unprivileged mode, *to cause execution* of the instruction within software having the privileged processing mode. Furthermore, the Ginsberg patent fails to teach or suggest *any* component within its disclosure that executes an indirect program flow control instruction when the current processing mode is unprivileged.

Additionally, the Examiner is correct in acknowledging that the Ginsberg patent is silent on the claimed step of determining a current processing mode of an executing software

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

function, recited in claim 1. (See 09/06/06 Office Action, p. 3, ¶ 4). However, the Examiner is incorrect in stating that the Wendorf patent cures this deficiency. The Examiner asserts that the testing as taught by the Wendorf patent to determine where the initiated thread is executing is equivalent to the determining a current processing mode of an executing software function, as recited in claim 1 of the present invention. (See *Id.*, p. 5, Response to Arguments). However, this assertion is incorrect. The determination of a memory location for a particular thread is not analogous to, nor equivalent to the determination of a mode of operation for a processor. The determination of a processing mode is not made on a thread-by-thread basis and the processing mode may not be dependant on the memory location for each thread. More specifically, a privileged processing mode may allow for the execution of a certain instruction, while an unprivileged processing mode may deny the execution of that very same instruction. Thus, it is respectfully submitted that the Wendorf patent does not disclose nor suggest, "determining a current processing mode of an executing software function" as recited in claim 1. Furthermore, similar to the Ginsberg patent, the Wendorf patent also fails to teach or suggest *any* component within its disclosure that executes an indirect program flow control instruction when the current processing mode is unprivileged.

Applicant respectfully submits that for at least the reasons stated above, claim 1 of the present application is not obvious over the Ginsberg patent in view of the Wendorf patent, and request that the rejection of this claim be withdrawn. As claims 2-5 depend from, and therefore include all the limitations of claim 1, it is hereby submitted that claims 2-5 are also allowable.

Claim 6 recites, *inter alia*, "identify a current processing mode of the program code segment" and "execute an indirect program flow control instruction if the current

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

processing mode is an unprivileged processing mode.” Thus, for the reasons described above with reference to claim 1, it is respectfully submitted that claim 6 is also allowable.

Claim 7 recites, *inter alia*, “identify a current processing mode of a program code segment” and “execute an indirect program flow control instruction if the current processing mode is an unprivileged processing mode.” Thus, for the reasons described above with reference to claim 1, it is respectfully submitted that claim 7 is also allowable.

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

8. Conclusions

RECEIVED
CENTRAL FAX CENTER

MAR 23 2007

For the reasons set forth above, Appellant respectfully requests that the Board reverse the final rejections of the claims by the Examiner under 35 U.S.C. § 103(a), and indicate that claims 1-7 are allowable.

Respectfully submitted,

Date: March 23, 2007

By: 
Michael J. Marcin (Reg. No. 48,198)

Fay Kaplun & Marcin, LLP
150 Broadway, Suite 702
New York, NY 10038
Tel: (212) 619-6000
Fax: (212) 619-0276

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

CLAIMS APPENDIX

RECEIVED
CENTRAL FAX CENTER

MAR 23 2007

1. A method, comprising:

determining a current processing mode of an executing software function;

when the current processing mode is a privileged processing mode, executing a direct program flow control instruction to directly access an instruction within a software having the privileged processing mode; and

when the current processing mode is an unprivileged processing mode, executing an indirect program flow control instruction to cause execution of the instruction within the software having the privileged processing mode.
2. The method of claim 1, wherein the direct program flow control instruction is a jump instruction.
3. The method of claim 1, wherein the indirect program flow control instruction is an interrupt instruction.
4. The method of claim 1, wherein the software having the privileged processing mode is operating system software.
5. The method of claim 4, wherein the software having the privileged processing mode is kernel software.

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

6. A method, comprising:

identifying a program code segment implementing an access to a memory area to be executed within a privileged processing mode;

replacing the program code segment with a substitute code segment;

wherein the substitute code segment includes program code to

identify a current processing mode of the program code segment,

execute a direct program flow control instruction if the current processing mode is the privileged processing mode, and

execute an indirect program flow control instruction if the current processing mode is an unprivileged processing mode.

7. A computer readable medium encoded with a software application, comprising:

a software code implementing application functionality; and

a smart system call into an operating system;

wherein the smart system call comprises the software code to

identify a current processing mode of a program code segment,

execute a direct program flow control instruction if the current processing mode is a privileged processing mode, and

execute an indirect program flow control instruction if the current processing mode is an unprivileged processing mode.

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

EVIDENCE APPENDIX

No evidence has been entered or relied upon in the present appeal.

RECEIVED
CENTRAL FAX CENTER
MAR 23 2007

Serial No.: 09/920,995
Group Art Unit: 2194
Attorney Docket No.: 40101 / 08201

RELATED PROCEEDING APPENDIX

No decisions have been rendered regarding the present appeal or any proceedings related thereto.

RECEIVED
CENTRAL FAX CENTER
MAR 23 2007